IN THE CLAIMS:

The text of all pending claims, (including withdrawn claims) is set forth below. Cancelled and not entered claims are indicated with claim number and status only. The claims as listed below show added text with underlining and deleted text with ~~strikethrough~~. The status of each claim is indicated with one of (original), (currently amended), (cancelled), (withdrawn), (new), (previously presented), or (not entered).

Please CANCEL claims 18-34 without prejudice or disclaimer in accordance with the following:

1. (PREVIOUSLY PRESENTED) A memory access method for a multiprocessor system which includes a plurality of system modules coupled via a crossbar module, each of the system modules including a buffer which holds data and a plurality of processors having a cache memory which temporarily holds data, said memory access method comprising:

responsive to a read request from a processor within an arbitrary system module, holding data preread from a system module, other than the arbitrary system module, in a buffer within the crossbar module.

2. (PREVIOUSLY PRESENTED) The memory access method as claimed in claim 1, further comprising:

setting information indicating whether or not to carry out a data preread with respect to the arbitrary system module, depending on a program which is executed by one or a plurality of processors within the arbitrary system module.

3. (PREVIOUSLY PRESENTED) The memory access method as claimed in claim 2, further comprising:

adding, to a data transfer of the preread data, a priority which is lower than a priority of a normal data transfer.

4. (PREVIOUSLY PRESENTED) The memory access method as claimed in claim 1, further comprising:

adding, to a data transfer of the preread data, a priority which is lower than a priority of a normal data transfer.

5. (PREVIOUSLY PRESENTED) A multiprocessor system, comprising:

a plurality of system modules;

at least one crossbar module; and

a bus coupling the system modules and the crossbar module,

each of the system modules including a buffer which holds data, a plurality of processors each having a cache memory which temporarily holds data, and a control unit which controls input and output of data with respect to the system module to which the control unit belongs,

a data transfer between two system modules being made via the crossbar module, and

said crossbar module including a buffer which holds data preread from a system module, other than an arbitrary system module, responsive to a read request from a processor within the arbitrary system module.

6. (PREVIOUSLY PRESENTED) The multiprocessor system as claimed in claim 5, wherein the arbitrary system module includes a unit setting information indicating whether or not to carry out a data preread with respect to the arbitrary system module, depending on a program which is executed by one or a plurality of processors within the arbitrary system module.

7. (PREVIOUSLY PRESENTED) The multiprocessor system as claimed in claim 6, wherein each of the system modules further includes a unit adding, to a data transfer of the preread data, a priority which is lower than a priority of a normal data transfer.

8. (PREVIOUSLY PRESENTED) The multiprocessor system as claimed in claim 5, wherein each of the system modules further includes a unit adding, to a data transfer of the preread data, a priority which is lower than a priority of a normal data transfer.

9. (PREVIOUSLY PRESENTED) The multiprocessor system as claimed in claim 5, wherein one of the system modules, which has a memory with a requested address of the read request, includes a unit starting a data preread at a timing before detecting a state of the cache memory included therein.

11. (PREVIOUSLY PRESENTED) A multiprocessor system, comprising:

a plurality of nodes each including a plurality of system modules, a crossbar module, and a bus coupling the system modules and the crossbar module within each node; and

a bus coupling adjacent nodes via the crossbar modules of the adjacent nodes,

each of the system modules including a buffer which holds data, a plurality of processors each having a cache memory which temporarily holds data, and a control unit which controls input and output of data with respect to the system module to which the control unit belongs,

a data transfer between two system modules being made via at least one crossbar module,

said crossbar module including a buffer which holds data preread from a system module, other than an arbitrary system module, in responsive to a read request from a processor within the arbitrary system module unit which controls input and output of data with respect to the system module to which the control unit belongs,

a data transfer between two system modules being made via at least one crossbar module, and

said crossbar module including a buffer which holds data preread from a system module, other than an arbitrary system module, responsive to a read request from a processor within the arbitrary system module.

12. (PREVIOUSLY PRESENTED) The memory access method as claimed in claim 1, wherein said holding data further comprises:

generating the read request from the arbitrary system module if a mishit occurs in the arbitrary system module;

prereading the data requested by the read request in the system module, other than the arbitrary system module, and storing the requested data; and

transferring the preread data from the system module, other than the arbitrary system module, to the buffer within the crossbar module, with a priority lower than a priority of a normal data transfer between the system modules and the crossbar module.

13. (PREVIOUSLY PRESENTED) The memory access method as claimed in claim 12, further comprising:

transferring the preread data stored in the buffer within the crossbar module to the arbitrary system module with a priority lower than the priority of the normal data transfer between the system modules and the crossbar module.

14. (PREVIOUSLY PRESENTED) The multiprocessor system as claimed in claim 5, wherein:

the arbitrary system module generates the read request if a mishit occurs in the arbitrary system module;

the system module, other than the arbitrary system module, storing the data requested by the read request prereads the requested data: and

the system module, other than the arbitrary system module, transfers the preread data to the buffer within the crossbar module, with a priority lower than a priority of a normal data transfer between the system modules and the crossbar module.

15. (PREVIOUSLY PRESENTED) The multiprocessor system as claimed in claim 14, wherein the crossbar module transfers the preread data stored in the buffer thereof to the arbitrary system module with a priority lower than the priority of the normal data transfer between the system modules and the crossbar module.

16. (PREVIOUSLY PRESENTED) The multiprocessor system as claimed in claim 11, wherein:

the arbitrary system module generates the read request if a mishit occurs in the arbitrary system module;

the system module, other than the arbitrary system module, storing the data requested by the read request prereads the requested data: and

the system module, other than the arbitrary system module, transfers the preread data to the buffer within the crossbar module, with a priority lower than a priority of a normal data transfer between the system modules and the crossbar module.

17. (PREVIOUSLY PRESENTED) The memory access method as claimed in claim 16, wherein the crossbar module transfers the preread data stored in the buffer thereof to the arbitrary system module with a priority lower than the priority of the normal data transfer between the system modules and the crossbar module.

18-34 CANCELLED